

# Nonconvex proximal splitting: batch and incremental algorithms\*

SUVRIT SRA

Max Planck Institute for Intelligent Systems  
72076 Tübingen, Germany  
suvrit@tuebingen.mpg.de

September 2, 2011

## Abstract

Within the unmanageably large class of nonconvex optimization, we consider the rich subclass of nonsmooth problems that have *composite* objectives—this already includes the extensively studied convex, composite objective problems as a special case. For this subclass, we introduce a powerful, new framework that permits asymptotically *non-vanishing* perturbations. In particular, we develop perturbation-based batch and incremental (online like) nonconvex proximal splitting algorithms. To our knowledge, this is the first time that such perturbation-based nonconvex splitting algorithms are being proposed and analyzed. While the main contribution of the paper is the theoretical framework, we complement our results by presenting some empirical results on matrix factorization.

## 1 Introduction

Within the unmanageably vast class of nonconvex optimization, we consider the rich subclass of problems that have *nonconvex composite objectives*. Specifically, we study problems of the form

$$\text{minimize } F(x) + \psi(x), \quad \text{s.t. } x \in \mathcal{X}, \quad (1)$$

where  $\mathcal{X} \subset \mathbb{R}^n$  is a compact convex set,  $F : \mathbb{R}^n \rightarrow \mathbb{R}$  is a differentiable function, and  $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$  is a lower semi-continuous (lsc) convex function. We make the common assumption that  $F \in C_L^1(\mathcal{X})$ , i.e., the gradient  $\nabla F$  is (locally) *Lipschitz continuous* on  $\mathcal{X}$  with constant  $L$ ,

$$\|\nabla F(x) - \nabla F(y)\| \leq L\|x - y\| \quad \text{for all } x, y \in \mathcal{X}. \quad (2)$$

Problem (1) is a natural but far-reaching generalization of *composite objective* convex problems that enjoy tremendous importance in machine learning; see [Duchi and Singer, 2009, Bach et al., 2011, Schmidt et al., 2009, Beck and Teboulle, 2009], for example. Although, convex formulations are extremely useful, often for many difficult problems a nonconvex formulation is more natural. Familiar examples include, matrix factorization [Mairal et al., 2010, Lee and Seung, 2000], blind deconvolution [Kundur and Hatzinakos, 1996], dictionary learning [Kreutz-Delgado et al., 2003, Mairal et al., 2010], and neural networks [Mangasarian, 1993, Haykin, 1994].

---

\*Reformatted, slightly enhanced version of MPI for Intelligent Systems Technical Report MPI-IS-2.

The main contribution of our paper is a new framework NOCOPS (an acronym for nonconvex proximal splitting). This framework solves (1) while allowing *computational errors*, a capability that proves key to deriving scalable batch and incremental (online like). A realistic feature of our framework is that it does *not* require the computational errors to vanish in the limit or its stepsizes to shrink to zero; such choices that are often assumed in standard analysis of (convex) incremental-gradient like methods [Bertsekas, 2010] or even in stochastic gradient methods [Gaivoronski, 1994].

NOCOPS builds on the remarkable work of Solodov [1997], but it is strictly *more general* than [Solodov, 1997] (which solves (1) only for  $\psi \equiv 0$ ). Like Solodov [1997]’s framework, NOCOPS also allows non-vanishing errors, which is practical, since often one has limited or no true control over computational errors (e.g., fixed noise level in a simulation). To our knowledge, ours is the first work on nonconvex proximal splitting that has *both batch and incremental* variants, regardless of the ability to handle nonvanishing errors.

**Related Work.** Among batch nonconvex splitting methods, an early paper is [Fukushima and Mine, 1981]. More recently, in his pioneering paper on composite minimization, Nesterov [2007] solved (1) via a splitting-like algorithm. Fukushima and Mine [1981] ensured convergence by forcing monotonic descent (using line-search); Nesterov [2007] proved convergence (for the nonconvex case) by also ensuring monotonic descent. Even more recently, Attouch et al. [2010] introduced a powerful method based on Kurdyka-Łojasiewicz theory, though convergence again hinged on descent. This insistence on monotonic descent makes these methods unsuitable to obtaining incremental, stochastic, or online variants.

But there are some incremental and stochastic methods that do apply to (1), namely the generalized gradient-type algorithms of [Solodov and Zavriev, 1998] and stochastic generalized gradient methods of [Ermoliev and Norkin, 1997, 1998]. Both approaches are analogous to subgradient methods from convex optimization, and face similar difficulties. For example, as is well recognized (see [Nesterov, 2007, Duchi and Singer, 2009], e.g.), subgradient-style methods fail to exploit composite objectives. Moreover, they exhibit the effect of the regularizer only in the limit; for example, if  $\psi(x) = \|x\|_1$ , then the sparse solutions are obtained only in the limit, and intermediate iterates may be dense.

For convex problems, a powerful alternative to subgradient methods is offered by *proximal splitting* (see [Combettes and Pesquet, 2010] for a survey). These methods split (1) into smooth and non-smooth parts. The smooth part is handled as in gradient-projection while the nonsmooth part is handled via a proximity operator. Owing to their ability to effectively tackle the nonsmooth part, proximal methods become valuable in machine learning and related areas; see [Combettes and Pesquet, 2010, Beck and Teboulle, 2009, Bach et al., 2011, Duchi and Singer, 2009] and the references therein.

## 2 The Nocops Framework

We begin by defining the function  $g : \mathbb{R}^n \rightarrow \mathbb{R}$  to be the sum  $g(x) := \psi(x) + \delta(x|\mathcal{X})$ ,

where  $\delta(x|\mathcal{X})$  is the *indicator function* for the set  $\mathcal{X}$ . With this notation, the main problem of this paper is

$$\text{minimize}_{x \in \mathbb{R}^n} \quad \phi(x) := F(x) + g(x). \quad (3)$$

Next, we recall a definition central to our analysis.

**Definition 1** (Proximity operator). Let  $g : \mathbb{R}^n \rightarrow \mathbb{R}$  be lsc and convex. The *proximity operator* for

$g$ , indexed by scalar  $\eta > 0$ , is the nonlinear map [see e.g., Rockafellar and Wets, 1998, Def. 1.22]:

$$P_\eta^g : y \mapsto \operatorname{argmin}_{x \in \mathbb{R}^n} \left( g(x) + \frac{1}{2\eta} \|x - y\|^2 \right). \quad (4)$$

Proximity operators are key to forward-backward splitting [Combettes and Wajs, 2005], which, for convex  $F \in C_L^1(\mathcal{X})$  and appropriate stepsizes  $\eta_k$ , optimizes (1) by essentially iterating

$$x^{k+1} = P_{\eta_k}^g(x^k - \eta_k \nabla F(x^k)), \quad k = 0, 1, \dots \quad (5)$$

Our new framework NOCOPS introduces two powerful generalizations to (5). First, it allows  $F$  to be nonconvex, and second, it allows perturbations. Formally, NOCOPS performs the iteration

$$x^{k+1} = P_{\eta_k}^g(x^k - \eta_k \nabla F(x^k) + \eta_k \vartheta(x^k)), \quad k = 0, 1, \dots, \quad (6)$$

where the stepsizes  $\eta_k$  satisfy the standard bounds and conditions

$$c \leq \liminf_k \eta_k, \quad \limsup_k \eta_k \leq \min\{1, 2/L - c\}, \quad 0 < c < 1/L. \quad (7)$$

The *perturbation* term  $\eta_k \vartheta(x^k)$  in (6) represents the computational errors, which occur for example when only an approximation to the full gradient  $\nabla F(x)$  is available.

To make NOCOPS well-defined, following [Solodov, 1997], we also impose a mild restriction on the perturbations. Specifically, we assume that for all  $\eta$  smaller than a fixed value  $\bar{\eta}$ , it holds that

$$\eta \|\vartheta(x)\| \leq \bar{\epsilon}, \quad \text{for some fixed } \bar{\epsilon} \geq 0, \quad \text{and } \forall x \in \mathcal{X}. \quad (8)$$

Condition (8) is weaker than the typical vanishing error requirement  $\eta \|\vartheta(x^k)\| \rightarrow 0$  imposed by most analyses. Since nonvanishing errors are allowed, exact stationary points cannot not always be obtained, but appropriate *inexact stationary points* can be found. To formalize this, recall that a point  $x^* \in \mathbb{R}^n$  is stationary for (3), if and only if it satisfies the inclusion

$$0 \in \partial_C \phi(x^*) := \nabla F(x^*) + \partial g(x^*), \quad (9)$$

where  $\partial_C \phi$  denotes the Clarke (generalized) subdifferential [Clarke, 1983]. Inclusion (9) may be equivalently recast as the fixed-point condition

$$x^* = P_\eta^g(x^* - \eta \nabla F(x^*)), \quad \text{for } \eta > 0. \quad (10)$$

We use (10) to characterize approximate stationary points. Define thus the *proximal residual*

$$\rho(x) := x - P_1^g(x - \nabla F(x)), \quad (11)$$

so that for stationary  $x^*$ , the residual norm  $\|\rho(x^*)\| = 0$ . At point  $x$ , let the level of perturbation be given by  $\epsilon(x) \geq 0$ . We define a point  $\bar{x}$  to be  $\epsilon$ -stationary if the residual norm is bounded satisfies

$$\|\rho(\bar{x})\| \leq \epsilon(\bar{x}). \quad (12)$$

To control overall level of perturbation in the system, we require  $\epsilon(x) \geq \eta \|\vartheta(x)\|$ . Thus, intuitively, by letting  $\eta$  become small enough, we can obtain a stationary point of any desired accuracy.

## 2.1 Convergence analysis

Our analysis builds on the pioneering works of Nesterov [2007] and Solodov [1997]. But as mentioned, our problem and analysis are *strictly* more general. Specifically, in contrast to [Nesterov, 2007], we permit perturbations and *do not* rely on strict descent, and unlike [Solodov, 1997], we consider nonsmooth objective functions. Our analysis yields, to our knowledge, the first nonconvex proximal splitting algorithm with nonvanishing noise, and also the first nonconvex incremental proximal splitting algorithm, regardless of vanishing or nonvanishing nature of the noise.

We begin by recalling two simple facts without proof; the first is a classical descent lemma.

**Lemma 1** (Descent lemma [see e.g., Nesterov, 2004, Lemma 2.1.3]). *Let  $F \in C_L^1(\mathcal{X})$ . Then,*

$$|F(x) - F(y) - \langle \nabla F(y), x - y \rangle| \leq \frac{L}{2} \|x - y\|^2, \quad \forall x, y \in \mathcal{X}. \quad (13)$$

**Lemma 2** (Nonexpansivity [see e.g., Combettes and Wajs, 2005, Lemma 2.4]). *The operator  $P_\eta^g$  is nonexpansive, that is,*

$$\|P_\eta^g x - P_\eta^g y\| \leq \|x - y\|, \quad \forall x, y \in \mathbb{R}^n. \quad (14)$$

Next, we prove a useful monotonicity result about proximity operators.

**Lemma 3** (Monotonicity). *Let  $P_\eta \equiv P_\eta^g$ ; let  $y, z \in \mathbb{R}^n$ , and  $\eta > 0$ , and define*

$$p(\eta) := \eta^{-1} \|P_\eta(y - \eta z) - y\|, \quad (15)$$

$$q(\eta) := \|P_\eta(y - \eta z) - y\|. \quad (16)$$

*Then,  $p(\eta)$  is a decreasing function of  $\eta$ , while  $q(\eta)$  is an increasing function of  $\eta$ .*

*Proof.* Both (15) and (16) essentially follow from properties of Moreau-envelopes [Rockafellar and Wets, 1998, Combettes and Wajs, 2005] (also see [Nesterov, 2007]). To set our notation, we provide a proof in the language of proximity operators. Consider thus the “deflected” proximity objective

$$m_g(x, \eta; y, z) := \langle z, x - y \rangle + \frac{1}{2} \eta^{-1} \|x - y\|^2 + g(x), \quad (17)$$

to which we associate the (deflected) *Moreau-envelope*

$$E_g(\eta) := \inf_{x \in \mathcal{X}} m_g(x, \eta; y, z). \quad (18)$$

Function  $m_g$  is easily seen to be a convex (see e.g., [Rockafellar and Wets, 1998, Theorem 2.26]); thus it follows that the infimum in (18) is attained at the unique point  $P_\eta^g(y - \eta z)$ . Thus,  $E_g(\eta)$  is differentiable, and

$$E_g'(\eta) = -\frac{1}{2} \eta^{-2} \|P_\eta^g(y - \eta z) - y\|^2 = -\frac{1}{2} p(\eta)^2.$$

Since  $E_g(\eta)$  is convex ([Rockafellar and Wets, 1998, Theorem 2.26]),  $E_g'$  is increasing; equivalently  $p(\eta)$  is decreasing. Similarly, note that  $\hat{E}_g(\gamma) := E_g(1/\gamma)$  is concave in  $\gamma$ , as it is a pointwise (indexed by  $x$ ) infimum of functions linear in  $\gamma$  [Boyd and Vandenberghe, 2004, §3.2.3]. Thus, its derivative

$$\hat{E}_g'(\gamma) = \frac{1}{2} \|P_{1/\gamma}^g(x - \gamma^{-1} y) - x\|^2 = q(1/\gamma),$$

is a decreasing function of  $\gamma$ ; writing  $\eta = 1/\gamma$  completes the argument, and proves (16).  $\square$

**Remark 1.** The monotonicity results (15) and (16) complement similar monotonicity results for projection operators derived in [Gafni and Bertsekas, 1984, Lemma 1].

Now we analyze the difference  $\phi(x^k) - \phi(x^{k+1})$ ; specifically, we derive an inequality

$$\phi(x^k) - \phi(x^{k+1}) \geq h(x^k), \quad (19)$$

where the potential function  $h(x)$  depends on  $\|\rho(x)\|$  and  $\epsilon(x)$ . Note that the potential  $h(x)$  is allowed to take on negative values because we do not insist on monotonic descent. To simplify notation, let  $u \equiv x^{k+1}$ ,  $x \equiv x^k$ , and  $\eta \equiv \eta_k$ , so that update (6) becomes

$$u = P_\eta(x - \eta \nabla F(x) + \eta \vartheta(x)). \quad (20)$$

With this notation, we now have the following descent-like theorem.

**Theorem 1.** *Let  $x \in \mathcal{X}$ ,  $u, \eta$  be as in (20), and  $\epsilon(x) \geq \eta \|\vartheta(x)\|$ . Then we have the bound*

$$\phi(x) - \phi(u) \geq \frac{2-L\eta}{2\eta} \|u - x\|^2 - \frac{1}{\eta} \epsilon(x) \|u - x\|. \quad (21)$$

Moreover, if  $\phi'(u)$  is the subgradient  $\phi'(u) := \nabla F(u) + s \in \partial_C \phi(u)$ , then we also have the bound

$$\langle \phi'(u), x - u \rangle \geq \frac{1-L\eta}{\eta} \|u - x\|^2 - \epsilon(x) \|u - x\| \quad (22)$$

*Proof.* Consider the directional derivative  $\mathrm{d}m_g$  (of  $m_g$ , with respect to  $x$ , and in the direction  $w$ ), which satisfies at  $x = u$  the optimality condition

$$\mathrm{d}m_g(u, \eta; y, z)(w) = \langle z + \eta^{-1}(u - y) + s, w \rangle \geq 0, \quad s \in \partial g(u). \quad (23)$$

In (23), set  $z = \nabla F(x) - \vartheta(x)$ ,  $y = x$ , and  $w = x - u$ ; then, rearrange to obtain

$$\langle \nabla F(x) - \vartheta(x), u - x \rangle \leq \langle \eta^{-1}(u - x) + s, x - u \rangle. \quad (24)$$

By Lemma 1 we have

$$\phi(u) \leq F(x) + \langle \nabla F(x), u - x \rangle + \frac{L}{2} \|u - x\|^2 + g(u). \quad (25)$$

Adding and subtracting  $\vartheta(x)$  in (25), and then combining with (24) we further obtain

$$\begin{aligned} \phi(u) &\leq F(x) + \langle \nabla F(x) - \vartheta(x), u - x \rangle + \frac{L}{2} \|u - x\|^2 + g(u) + \langle \vartheta(x), u - x \rangle \\ &\leq F(x) + \langle \eta^{-1}(u - x) + s, x - u \rangle + \frac{L}{2} \|u - x\|^2 + g(u) + \langle \vartheta(x), u - x \rangle \\ &= F(x) + g(u) + \langle s, x - u \rangle + \left(\frac{L}{2} - \frac{1}{\eta}\right) \|u - x\|^2 + \langle \vartheta(x), u - x \rangle \\ &\leq F(x) + g(x) - \frac{2-L\eta}{2\eta} \|u - x\|^2 + \langle \vartheta(x), u - x \rangle \\ &\leq \phi(x) - \frac{2-L\eta}{2\eta} \|u - x\|^2 + \|\vartheta(x)\| \|u - x\| \\ &\leq \phi(x) - \frac{2-L\eta}{2\eta} \|u - x\|^2 + \frac{1}{\eta} \epsilon(x) \|u - x\|, \end{aligned}$$

where the third inequality follows from convexity of  $g$ , the fourth one from Cauchy-Schwarz, and the last one from the definition of  $\epsilon(x)$ . Flipping signs, we immediately obtain (23).

To prove (22), consider the directional derivative  $\mathrm{d}\phi(u; x - u)$ , for which using the identity  $\phi'(u) = \nabla F(u) + s$ , we have

$$\begin{aligned} \langle \nabla F(u) + s, x - u \rangle &= \langle \nabla F(x) - \vartheta(x) + s, x - u \rangle - \langle \nabla F(u) - \nabla F(x) + \vartheta(x), u - x \rangle \\ &\geq \langle \eta^{-1}(x - u), x - u \rangle - \langle \nabla F(u) - \nabla F(x), u - x \rangle - \langle \vartheta(x), u - x \rangle \\ &\geq (\eta^{-1} - L) \|u - x\|^2 - \langle \vartheta(x), u - x \rangle \\ &\geq \frac{1-L\eta}{\eta} \|u - x\|^2 - \|\vartheta(x)\| \|u - x\| \\ &\geq \frac{1-L\eta}{\eta} \|u - x\|^2 - \eta^{-1} \epsilon(x) \|u - x\|. \square \end{aligned}$$

Next, we bound the right-hand side terms in (21), by deriving two-sided bounds on  $\|x - u\|$ .

**Lemma 4.** *Let  $x$ ,  $u$ , and  $\eta$  be as in Theorem 1, and  $c$  be as defined in (7). Then,*

$$c\|\rho(x)\| - \eta^{-1}\epsilon(x) \leq \|x - u\| \leq \|\rho(x)\| + \eta^{-1}\epsilon(x). \quad (26)$$

*Proof.* From Lemma 3 we see that for  $\eta > 0$  it holds that

$$1 \leq \eta \implies q(1) \leq q(\eta), \quad \text{and} \quad 1 \geq \eta \implies p(1) \leq p(\eta) = \eta^{-1}q(\eta) \quad (27)$$

Using (27), the triangle inequality, and nonexpansivity (Lemma 2), we see that

$$\begin{aligned} \min\{1, \eta\} q(1) &= \min\{1, \eta\} \|\rho(x)\| \leq \|P_\eta(x - \eta \nabla f(x)) - x\| \\ &\leq \|x - u\| + \|u - P_\eta(x - \eta \nabla f(x))\| \\ &\leq \|x - u\| + \|\vartheta(x)\| \leq \|x - u\| + \eta^{-2}\epsilon(x). \end{aligned}$$

Since  $c \leq \liminf_k \eta_k$ , we have  $\|x - u\| \geq c\|\rho(x)\| - \epsilon(x)$ . An upper bound follows upon noting

$$\begin{aligned} \|x - u\| &\leq \|x - P_\eta(x - \eta \nabla f(x))\| + \|P_\eta(x - \eta \nabla f(x)) - u\| \\ &\leq \max\{1, \eta\} \|\rho(x)\| + \|\vartheta(x)\| \leq \|\rho(x)\| + \eta^{-1}\epsilon(x). \quad \square \end{aligned}$$

Theorem 1 and Lemma 4 have done the hard work; they imply the following crucial corollary.

**Corollary 1.** *Let  $x$ ,  $u$ ,  $\eta$ , and  $c$  be as in Lemma 4 and Theorem 1. Then, we have*

$$\phi(x) - \phi(u) \geq h(x),$$

where the function  $h$  is given by

$$h(x) := \frac{2-L\eta}{2\eta} c^2 \|\rho(x)\|^2 - \left(c \frac{2-L\eta}{\eta^2} + \frac{1}{\eta}\right) \|\rho(x)\| \epsilon(x) - \left(\frac{2-L\eta}{2\eta^3} + \frac{1}{\eta^2}\right) \epsilon(x)^2. \quad (28)$$

*Proof.* Plug in the bounds (26) into (21) and simplify.  $\square$

Now we are in a position to state our main convergence theorem.

**Theorem 2** (Convergence). *Let  $f \in C_L^1(\mathcal{X})$  such that  $\inf_{\mathcal{X}} f > -\infty$  and  $g$  be lsc, convex on  $\mathcal{X}$ . Let  $\{x^k\} \subset \mathcal{X}$  be a sequence generated by (6), and let condition (8) hold. Then, there exists a limit point  $x^*$  of the sequence  $\{x^k\}$ , and a constant  $K > 0$ , such that  $\|\rho(x^*)\| \leq K\epsilon(x^*)$ . Moreover, if the sequence  $\{F(x^k)\}$  converges, then for every limit point  $x^*$  of  $\{x^k\}$  it holds that  $\|r(x^*)\| \leq K\epsilon(x^*)$ .*

*Proof.* The hard work has been done by Theorem 1 and Lemma 4, because of which we have the bound  $\phi(x) - \phi(u) \geq h(x)$ , where the function  $h(x)$  is defined as

$$h(x) := \frac{2-L\eta}{2\eta} c^2 \|\rho(x)\|^2 - \left(c \frac{2-L\eta}{\eta} + 1\right) \|\rho(x)\| \epsilon(x) - \left(\frac{2-L\eta}{2\eta} + 1\right) \epsilon(x)^2. \quad (29)$$

Thus, Corollary 1 reduces the proof to a case where the arguments of [Solodov, 1997] become applicable; thus, we may conclude convergence; we omit details to avoid repetition.  $\square$

### 3 Incremental Nocops

We now specialize NOCOPS to the large-scale setting with decomposable  $F(x)$ , that is, to

$$\text{minimize} \quad (F(x) := \sum_{t=1}^T f_t(x)) + g(x), \quad (30)$$

where each  $f_t : \mathbb{R}^n \rightarrow \mathbb{R}$  is a  $C_{L_t}^1(\mathcal{X})$  function (let  $L_t \leq L$  for simplicity), and  $g, \mathcal{X}$  are as before.

In machine learning and optimization it has long been known that for decomposable objectives it can be advantageous to replace the full gradient  $\nabla F(x)$  by an *incremental gradient*  $\nabla f_{r(t)}(x)$ , where  $r(t)$  is some suitable index. Incremental methods have been extensively analyzed in the setting of backpropagation algorithms [Bertsekas, 2010, Solodov, 1997], a setting that corresponds to  $g(x) \equiv 0$  in our case. For  $g(x) \neq 0$ , the stochastic generalized gradient methods of [Ermoliev and Norkin, 1998] or the perturbed generalized methods of [Solodov and Zavriev, 1998] apply. But as previously argued, these methods suffer from problems similar to those faced by ordinary subgradient methods; so we may prefer proximal splitting methods instead.

To specialize NOCOPS for solving (30), we propose the following iteration:

$$\begin{aligned} x^{k+1} &= \mathcal{M}(x^k - \eta_k \sum_{t=1}^T \nabla f_t(z^t)) \\ z^1 &= x^k, \quad z^{t+1} = \mathcal{O}(z^t - \eta \nabla f_t(z^t)), \quad t = 1, \dots, T-1. \end{aligned} \quad (31)$$

Here,  $\mathcal{O}$  and  $\mathcal{M}$  are appropriate nonexpansive maps, choosing which we get different algorithms. For example, when  $\mathcal{X} = \mathbb{R}^n$ ,  $g(x) \equiv 0$ , and  $\mathcal{M} = \mathcal{O} = \text{Id}$ , then (31) reduces to the problem class considered in [Solodov, 1998]. If  $\mathcal{X}$  is a closed convex set,  $g(x) \equiv 0$ ,  $\mathcal{M} = \Pi_{\mathcal{X}}$ , and  $\mathcal{O} = \text{Id}$ , then (31) reduces to a method that is essentially implicit in [Solodov, 1998]. Note, however, that in this case, the constraints are enforced *only once* every major iteration; the minor iterates ( $z^t$ ) may be infeasible.

We introduce below four variants of (31); to our knowledge, *all four* are novel.

1.  $\mathcal{X} = \mathbb{R}^n$ ,  $g(x) \neq 0$ ,  $\mathcal{M} = P_{\eta}^g$ , and  $\mathcal{O} = \text{Id}$ ; this is a penalized unconstrained problem, and the penalty is applied *once* every major iteration.
2.  $\mathcal{X} = \mathbb{R}^n$ ,  $g(x) \neq 0$ ,  $\mathcal{M} = P_{\eta}^g$ , and  $\mathcal{O} = P_{\eta}^g$ ; this is a penalized unconstrained problem, but now the penalty is applied at every minor iteration.
3.  $\mathcal{X}$  is a closed convex set,  $g(x) = \psi(x) + \delta(\cdot|\mathcal{X})$  (where  $\psi$  may be zero or nonzero),  $\mathcal{M} = P_{\eta}^g$ , and  $\mathcal{O} = \text{Id}$ ; this is a penalized, constrained problem, and the penalty is applied once every major iteration.
4. Same as variant 3, except that  $\mathcal{O} = P_{\eta}^g$ ; this is a penalized, constrained problem, and the penalty is applied at every minor iteration.

Which of the four variants one prefers depends on the complexity of the constraint set  $\mathcal{X}$  and the regularizer  $g(x)$ . However, the analysis of all four variants is similar, so we present details only for the fourth, as it is the most general.

<p><b>Input:</b> <math>\{\nabla f_t(X)\}</math>, <math>P_\eta^g</math>: subroutines</p> <p><b>Output:</b> Approximate solution to (31)</p> <p><math>t \leftarrow 1</math>; <math>k \leftarrow 0</math>; <math>x^k \in \mathcal{X}</math>;</p> <p><b>while</b> <math>\neg \text{converged}</math> <b>do</b></p> <div style="margin-left: 20px;"> <p><math>z^1 \leftarrow x^k</math>; <math>\alpha_1 \leftarrow \eta_k</math>;</p> <p><b>if</b> <math>t &lt; T</math> <b>then</b></p> <div style="margin-left: 20px;"> <p>Get <math>t</math>-th gradient <math>g^t = \nabla f_t(z^t)</math>;</p> <p>Compute <math>z^{t+1} \leftarrow P_{\alpha_t}^g(z^t - \alpha_t g^t)</math>;</p> <p>Aggregate gradient <math>F^k \leftarrow F^k + g^t</math>;</p> <p><math>t \leftarrow t + 1</math>;    update stepsize <math>\alpha_t</math>;</p> </div> <p><b>else</b></p> <div style="margin-left: 20px;"> <p><math>x^{k+1} = P_{\eta_k}^g(x^k - \eta_k F^k)</math>;</p> <p><math>t \leftarrow 0</math>;    <math>k \leftarrow k + 1</math>;</p> <p>Check for convergence;</p> </div> </div>
---

**Algorithm 1:** Incremental NOCOPS.

### 3.1 Convergence

We begin by rewriting (31) in a form that matches the main iteration (6):

$$x^{k+1} = \mathcal{M}(x^k - \eta_k \sum_{t=1}^T \nabla f_t(z^t)) = \mathcal{M}(x^k - \eta_k \nabla F(x^k) + \eta_k \vartheta(x^k)),$$

where the error term at a general  $x$  is given by  $\vartheta(x) := \sum_{t=1}^T (f_t(x) - f_t(z^t))$ . We must ensure that the norm of the error term is bounded. Lemma 7 proves this bound; however, we first prove two helpful lemmas.

**Lemma 5** (Bounded increment). *Let  $z^{t+1}$  be computed by (31). Then, we have*

$$\text{if } \mathcal{O} = \text{Id, then } \|z^{t+1} - z^t\| = \eta \|\nabla f_t(z^t)\| \quad (32)$$

$$\text{if } \mathcal{O} = \Pi_{\mathcal{X}}, \text{ then } \|z^{t+1} - z^t\| \leq \eta \|\nabla f_t(z^t)\| \quad (33)$$

$$\text{if } \mathcal{O} = P_\eta^g \text{ and } s^t \in \partial g(z^t), \text{ then } \|z^{t+1} - z^t\| \leq 2\eta \|\nabla f_t(z^t) + s^t\|. \quad (34)$$

*Proof.* Relation (32) is obvious and (33) follows immediately from nonexpansivity of projections. For proving (34), notice that definition ((4)) implies the inequality

$$\begin{aligned} \frac{1}{2} \|z^{t+1} - z^t + \eta \nabla f_t(z^t)\|^2 + \eta g(z^{t+1}) &\leq \frac{1}{2} \|\eta \nabla f_t(z^t)\|^2 + \eta g(z^t), \\ \frac{1}{2} \|z^{t+1} - z^t\|^2 &\leq \eta \langle \nabla f_t(z^t), z^t - z^{t+1} \rangle + \eta (g(z^t) - g(z^{t+1})). \end{aligned}$$

But since  $\psi$  is convex, we know that

$$g(z^{t+1}) \geq g(z^t) + \langle s_t, z^{t+1} - z^t \rangle, \quad s_t \in \partial g(z^t).$$

Since  $g$  is convex it further follows that

$$\begin{aligned} \frac{1}{2} \|z^{t+1} - z^t\|^2 &\leq \eta \langle s^t, z^t - z^{t+1} \rangle + \langle \nabla f_t(z^t), z^t - z^{t+1} \rangle \\ &\leq \eta \|s_t + \nabla f_t(z^t)\| \|z^t - z^{t+1}\| \\ \implies \|z^{t+1} - z^t\| &\leq 2\eta \|\nabla f_t(z^t) + s^t\|. \end{aligned}$$

□



**Lemma 6** (Incrementality error). *Let  $x \equiv x^k$ , and define*

$$\epsilon_t := \|\nabla f_t(z^t) - \nabla f_t(x)\|, \quad t = 1, \dots, T. \quad (35)$$

*Then, for each  $t \geq 2$ , the following bound on the error holds:*

$$\epsilon_t \leq 2\eta L \sum_{j=1}^{t-1} (1 + 2\eta L)^{t-1-j} \|\nabla f_j(x) + s^j\|, \quad t = 2, \dots, T. \quad (36)$$

*Proof.* The proof extends the unconstrained, unpenalized setting of [Solodov, 1998] to our setting. We proceed by induction. The base case is  $t = 2$ , for which we have

$$\epsilon_2 = \|\nabla f_2(z^2) - \nabla f_2(x)\| \leq L\|z^2 - x\| = L\|z^2 - z^1\| \stackrel{(34)}{\leq} 2\eta L \|\nabla f_1(x) + s^1\|.$$

Assume inductively that (36) holds for  $t \leq r < T$ , and consider  $t = r + 1$ . In this case we have

$$\begin{aligned} \epsilon_{r+1} &= \|\nabla f_{r+1}(z^{r+1}) - \nabla f_{r+1}(x)\| && \leq L\|z^{r+1} - x\| \\ &= L \left\| \sum_{j=1}^r (z^{j+1} - z^j) \right\| && \leq L \sum_{j=1}^r \|z^{j+1} - z^j\| \\ &\stackrel{\text{Lemma 5}}{\leq} 2\eta L \sum_{j=1}^r \|\nabla f_j(z^j) + s^j\|. \end{aligned} \quad (37)$$

To complete the induction, first observe that  $\|\nabla f_t(z^t) + s^t\| \leq \|\nabla f_t(x) + s^t\| + \epsilon_t$ . Thus, invoking the induction hypothesis, we obtain

$$\|\nabla f_t(z^t)\| \leq \|\nabla f_t(x)\| + 2\eta L \sum_{j=1}^{t-1} (1 + 2\eta L)^{t-1-j} \|\nabla f_j(x) + s^j\|, \quad t = 2, \dots, r. \quad (38)$$

Combining inequality (38) with (37) we further obtain

$$\epsilon_{r+1} \leq 2\eta L \sum_{j=1}^r \left( \|\nabla f_j(x) + s^j\| + 2\eta L \sum_{l=1}^{j-1} (1 + L\eta)^{j-1-l} \|\nabla f_l(x) + s^l\| \right).$$

Introducing the shorthand  $\beta_j \equiv \|\nabla f_j(x) + s^j\|$ , simple manipulation of the above inequality yields

$$\begin{aligned} \epsilon_{r+1} &\leq 2\eta L \beta_r + \sum_{l=1}^{r-1} \left( 2\eta L + 4\eta^2 L^2 \sum_{j=l+1}^r (1 + 2\eta L)^{j-l-1} \right) \beta_l \\ &= 2\eta L \beta_r + \sum_{l=1}^{r-1} \left( 2\eta L + 4\eta^2 L^2 \sum_{j=0}^{r-l-1} (1 + 2\eta L)^j \right) \beta_l \\ &= 2\eta L \beta_r + \sum_{l=1}^{r-1} 2\eta L (1 + 2\eta L)^{r-l} \beta_l = 2\eta L \sum_{l=1}^r (1 + 2\eta L)^{r-l} \beta_l, \end{aligned}$$

which completes the proof.  $\square$

Now we are ready to bound the error using Lemma 7.

**Lemma 7** (Bounded error). *If for all  $x \in \mathcal{X}$ ,  $\|\nabla f_t(x)\| \leq M$  and  $\|\partial g(x)\| \leq G$ , then  $\|\vartheta(x)\| \leq K_1$ , for some constant  $K_1 > 0$ .*

*Proof.* First, observe that if  $z^{t+1}$  is computed by (31),  $\mathcal{O} = P_\eta^g$ , and  $s^t \in \partial g(z^t)$ , then

$$\|z^{t+1} - z^t\| \leq 2\eta \|\nabla f_t(z^t) + s^t\|. \quad (39)$$

Using (39) we can bound the error incurred upon using  $z^t$  instead of  $x^k$ . Specifically, if  $x \equiv x^k$ , and

$$\epsilon_t := \|\nabla f_t(z^t) - \nabla f_t(x)\|, \quad t = 1, \dots, T, \quad (40)$$

then Lemma 6 shows the following bound:

$$\epsilon_t \leq 2\eta L \sum_{j=1}^{t-1} (1 + 2\eta L)^{t-1-j} \|\nabla f_j(x) + s^j\|, \quad t = 2, \dots, T. \quad (41)$$

Since  $\epsilon_1 = 0$ , we have

$$\begin{aligned} \|\vartheta(x)\| &\leq \sum_{t=2}^T \epsilon_t \stackrel{(41)}{\leq} 2\eta L \sum_{t=2}^T \sum_{j=1}^{t-1} (1 + 2\eta L)^{t-1-j} \beta_j \\ &= 2\eta L \sum_{t=1}^{T-1} \beta_t \left( \sum_{j=0}^{T-t-1} (1 + 2\eta L)^j \right) = \sum_{t=1}^{T-1} \beta_t ((1 + 2\eta L)^{T-t} - 1) \\ &\leq \sum_{t=1}^{T-1} (1 + 2\eta L)^{T-t} \beta_t \leq (1 + 2\eta L)^{T-1} \sum_{t=1}^{T-1} \|\nabla f_t(x) + s^t\| \\ &\leq C_1(T-1)(M+G) =: K_1. \quad \square \end{aligned}$$

**Remark 2.** Lemma 7 implies that if in the error condition (8), we let  $\eta \rightarrow 0$ , then  $\eta\|\vartheta(x)\| \rightarrow 0$ .

Given the error bounds established by Lemma 7, convergence results for Algorithm 1 follow immediately from the more the general Theorem 2; we omit details to avoid repetition.

## 4 Application

The main contribution and focus of this paper is the new NOCOPS framework, and studying a specific application is not one of the aims of this paper. Nevertheless, we do illustrate NOCOPS's empirical performance on a challenging nonconvex problem, namely, (penalized) nonnegative matrix factorization:

$$\min_{X, A \geq 0} \quad \frac{1}{2} \|Y - XA\|_F^2 + \psi_0(X) + \sum_{t=1}^T \psi_t(a_t), \quad (42)$$

where  $Y$  is an  $m \times T$  matrix,  $X$  is  $m \times K$ , and  $A$  is  $K \times T$  with  $a_1, \dots, a_T$  as its columns. Problem (42) extends the famous nonnegative matrix factorization (NMF) problem [Lee and Seung, 2000] by allowing  $Y$  to be arbitrary (not necessarily nonnegative) and adding nonsmooth regularizers on  $X$  and  $A$ .

A similar class of problems was recently also studied in [Mairal et al., 2010], but with a crucial difference: the formulation in [Mairal et al., 2010] does not allow nonsmooth regularizers on  $X$  (the class of problems studied in [Mairal et al., 2010] is a subset of those our framework allows). On a technical note, [Mairal et al., 2010] consider stochastic optimization methods whose analysis requires perturbations to disappear in the limit; while our method is deterministic and our analysis does not rely on disappearing perturbations.

Following [Mairal et al., 2010] we rewrite (42) in a form more amenable to NOCOPS, that is,

$$\min_X \quad \phi(X) := \sum_{t=1}^T f_t(X) + g(X), \quad (43)$$

where  $g(X)$  captures both  $\psi_0(X)$  and the constraints on  $X$ . Each  $f_t(X)$  is defined as

$$f_t(X) := \min_a \quad \frac{1}{2} \|y_t - Xa\|^2 + g_t(a), \quad 1 \leq t \leq T, \quad (44)$$

where  $g_t(a)$  captures both  $\psi_t(a)$  and the constraints on  $a_t$ . Whenever (44) attains its unique minimum, say  $a^*$ , then  $f_t(X)$  is differentiable and we have  $\nabla_X f_t(X) = (y_t - Xa^*)(a^*)^T$ . Thus, we can instantiate Algorithm 1; all we need is a subroutine for solving (44).<sup>1</sup>

In our experiments, we consider the following two instances of (43): (i)  $g(X) = \delta(X \geq 0)$ ; and (ii)  $g(X) = \lambda \|X\|_1 + \delta(X \geq 0)$ . We select the  $g_t$ 's to be matching, so that  $g_t = \delta(a \geq 0)$ , and  $g_t(a) = \gamma \|a\|_1 + \delta(a \geq 0)$  are used. Choice (i) solves standard NMF, while choice (ii) solves a sparse-NMF problem. We remark that in general, when penalizing  $X$  one should either constrain  $a_t$  or penalize it, otherwise one can get degenerate solutions.

To provide the reader with a baseline, on the basic NMF problem we compare NOCOPS against the well-tuned C++ toolbox SPAMS [Mairal et al., 2010]. Obviously, the comparisons are not fair to NOCOPS, because unlike SPAMS, it is implemented in MATLAB. Fortunately, our MATLAB implementation already runs very competitively, and unlike SPAMS, also allows factorizing sparse matrices. We note that since our subroutines depend heavily on matrix-vector operations, a well-tuned C++ implementation of NOCOPS should run at least 3-4 times (based on initial experiments) faster than our MATLAB version, especially for sparse matrices.

*Remark:* The speed comparisons should not detract the reader from the main message: NOCOPS offers a new proximal operator based optimization framework that runs very competitively. With careful implementation, one can expect to see huge gains in speed, but engineering such a careful implementation is not one of the main points of this paper.

We compute NMF on the following data matrices:

1. CBCL Face Database [Sung, 1996] (dense, size  $361 \times 2429$ ); we compute a rank-49 factorization.
2. Yale B Database [Lee et al., 2005]; (dense, size  $32256 \times 2414$ ); we compute a rank-64 factorization.
3. Random matrix (dense, size  $4000 \times 4000$ , entries in  $[0, 1]$ ); we compute a rank-64 factorization, and penalize  $X$  by  $\lambda \|\cdot\|_1$ , and  $A$  by  $\gamma \|A\|_1$ , with  $(\lambda, \gamma) = (10^2, 10^{-4})$ .
4. Pajek connectivity matrix for Internet routers (sparse, size  $124,651 \times 124,651$ , density  $1.3 \cdot 10^{-5}$ , from the UFL sparse matrix collection, ID: 1505 [Davis and Hu, 2011]); we compute a rank-4 factorization; here  $(\lambda, \gamma) = (10^{-2}, 10^{-6})$  were used.

Figure 1 reports summarizes our experimental results. In the first row, in addition to SPAMS, we include running times for Lee and Seung's algorithm, and our implementation of alternating (nonnegatively constrained) least squares. From the graph we see that our MATLAB implementation of NOCOPS runs only slightly slower than the state-of-the-art method in SPAMS. The plots also show a dashed line that hints at what might be achievable with a faster C++ implementation of NOCOPS.

The second row in Figure 1 shows numerical results that compare the stochastic generalized gradient (SGGD) algorithm of [Ermoliev and Norkin, 1998] against NOCOPS, when started at exactly the same point. As is well-known, SGGD requires careful stepsize tuning to be competitive. Thus, we searched over a range of possible stepsize choices, and have reported the results with the best choices found. NOCOPS also requires some stepsize tuning, but significantly lesser than SGGD. Finally, we note that as predicted, the solutions returned by NOCOPS, often have objective function values better than SGGD, and always achieved greater sparsity.

## 5 Discussion

We presented a new framework called NOCOPS, which solves a broad class of nonconvex composite objective problems. NOCOPS builds on the general analysis of [Solodov, 1997], and extends it to

---

<sup>1</sup>In practice, it is better to use *mini-batches*, and we used them for all the online algorithms compared.

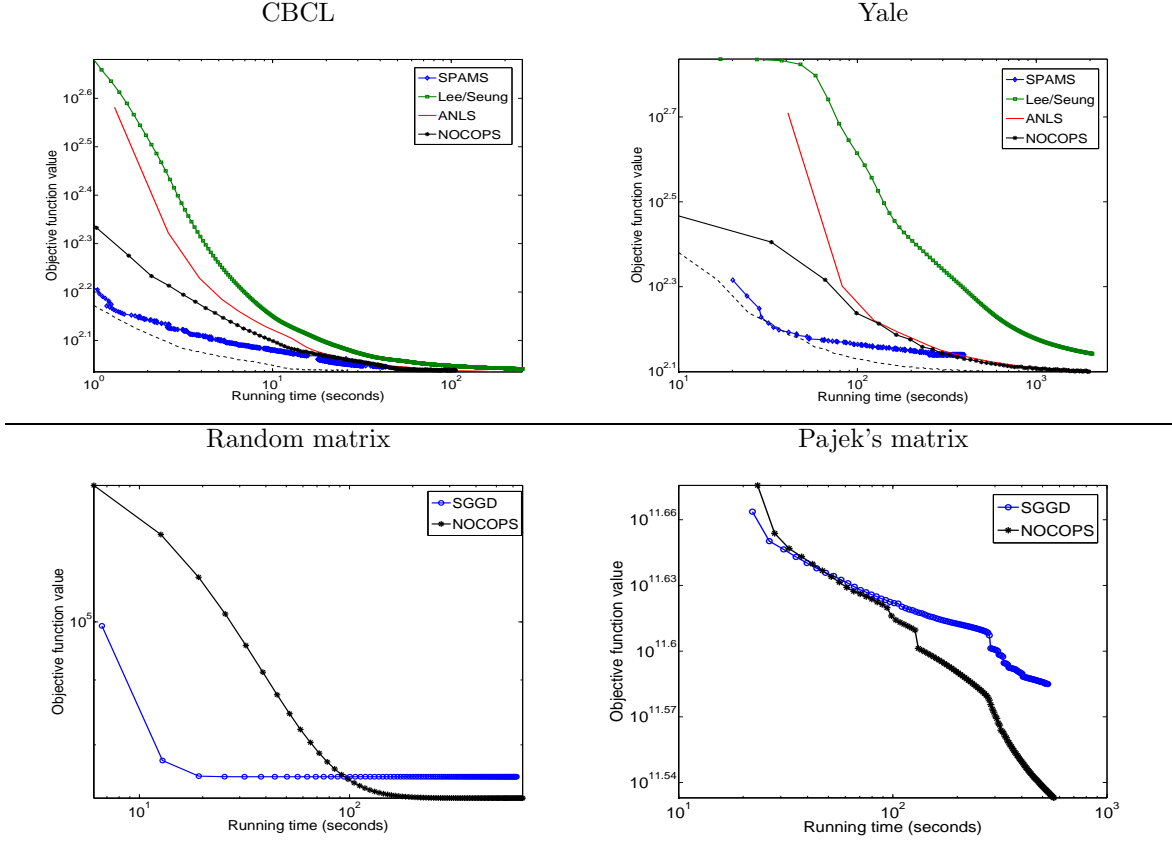


Figure 1: Top row: running times on CBCL and Yale B data. Bottom line, SGGD against NOCOPS. The densities of solutions returned (left to right) were (100%, 58.8%) for SGGD, and (61.1%, 0.1%) for NOCOPS. Initial objective values and very small runtimes have been suppressed for clarity of presentation.

admit problems that are *strictly* more general. NOCOPS permits nonvanishing perturbations, which is a useful practical feature. We exploited the perturbation analysis to derive both batch and incremental versions of NOCOPS. Finally, experiments with medium to large matrices showed that NOCOPS is competitive with state-of-the-art methods; NOCOPS was also seen to outperform the stochastic generalized gradient method.

We conclude by mentioning NOCOPS includes numerous algorithms and problem settings as special cases. Example are: forward-backward splitting with convex costs, incremental forward-backward splitting (convex), gradient projection (both convex and nonconvex), the proximal-point algorithm, and so on. Thus, it will be valuable to investigate if some of the theoretical results for these methods can be carried over to NOCOPS. Theoretically, the most important open problem that we would like to analyze is to permit even the regularizer in (1) to be nonconvex—but this might require significantly different convergence analysis.

## References

- H. Attouch, J. Bolte, and B. F. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Optimization Online*, Dec. 2010.
- F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Convex optimization with sparsity-inducing norms. In S. Sra, S. Nowozin, and S. J. Wright, editors, *Optimization for Machine Learning*. MIT Press, 2011.
- A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM J. Imaging Sciences*, 2(1):183–202, 2009.
- D. P. Bertsekas. Incremental Gradient, Subgradient, and Proximal Methods for Convex Optimization: A Survey. Technical Report LIDS-P-2848, MIT, August 2010.
- S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004. ISBN 0521833787.
- F. H. Clarke. *Optimization and nonsmooth analysis*. John Wiley & Sons, Inc., 1983.
- P. L. Combettes and J.-C. Pesquet. Proximal Splitting Methods in Signal Processing. *arXiv:0912.3522v4*, May 2010.
- P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200, 2005.
- T. A. Davis and Y. Hu. The University of Florida Sparse Matrix Collection. *ACM Transactions on Mathematical Software*, 2011. To appear.
- J. Duchi and Y. Singer. Online and Batch Learning using Forward-Backward Splitting. *J. Mach. Learning Res. (JMLR)*, Sep. 2009.
- Y. M. Ermoliev and V. I. Norkin. Stochastic generalized gradient method with application to insurance risk management. Technical Report IR-97-021, IIASA, Austria, April 1997.
- Y. M. Ermoliev and V. I. Norkin. Stochastic generalized gradient method for nonconvex nonsmooth stochastic optimization. *Cybernetics and Systems Analysis*, 34:196–215, 1998.
- M. Fukushima and H. Mine. A generalized proximal point algorithm for certain non-convex minimization problems. *Int. J. Systems Science*, 12(8):989–1000, 1981.
- E. M. Gafni and D. P. Bertsekas. Two-metric projection methods for constrained optimization. *SIAM Journal on Control and Optimization*, 22(6):936–964, 1984.
- A. A. Gaivoronski. Convergence properties of backpropagation for neural nets via theory of stochastic gradient methods. Part 1. *Optimization methods and Software*, 4(2):117–134, 1994.
- S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, 1st edition, 1994.
- K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural Computation*, 15:349–396, 2003.
- D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE Signal Processing Magazine*, 13(3):43–64, may 1996.
- D. D. Lee and H. S. Seung. Algorithms for nonnegative matrix factorization. In *NIPS*, pages 556–562, 2000.
- K.C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 27(5):684–698, 2005.
- J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online Learning for Matrix Factorization and Sparse Coding. *JMLR*, 11:10–60, 2010.
- O. L. Mangasarian. Mathematical Programming in Neural Networks. *Inform. J. Computing*, 5(4):

- 349–360, 1993.
- Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2004.
- Y. Nesterov. Gradient methods for minimizing composite objective function. Technical Report 2007/76, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), September 2007.
- R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*. Springer, 1998.
- M. Schmidt, E. van den Berg, M. Friedlander, and K. Murphy. Optimizing Costly Functions with Simple Constraints: A Limited-Memory Projected Quasi-Newton Algorithm. In *AISTATS*, 2009.
- M. V. Solodov. Convergence analysis of perturbed feasible descent methods. *J. Optimization Theory and Applications*, 93(2):337–353, 1997.
- M. V. Solodov. Incremental gradient algorithms with stepsizes bounded away from zero. *Computational Optimization and Applications*, 11:23–35, 1998.
- M. V. Solodov and S. K. Zavriev. Error stability properties of generalized gradient-type algorithms. *J. Optimization Theory and Applications*, 98(3):663–680, 1998.
- K.-K. Sung. *Learning and Example Selection for Object and Pattern Recognition*. PhD thesis, MIT, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Cambridge, MA, 1996.

## A Implementation notes

If  $\mathcal{X}$  is the nonnegative orthant  $\mathbb{R}_+^n$ , then the proximity operator  $P_\eta^g$  often simplifies as

$$P_\eta^g(y) = P_\eta^\psi(\Pi_{\mathcal{X}}y). \quad (45)$$

Additionally, if  $\psi$  is an elementwise separable function, then one can easily admit a box-plus-hyperplane constraint set  $\mathcal{X}$  of the form

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid l_i \leq x_i \leq u_i, \text{ for } 1 \leq i \leq n, \text{ and } a^T x = b\}. \quad (46)$$

For more general constraint sets, we can invoke *Dykstra splitting* [Combettes and Pesquet, 2010], which solves the problem

$$\min \quad \frac{1}{2} \|x - y\|^2 + \psi(x) + \delta(x|\mathcal{X}), \quad (47)$$

by using the following algorithm

<i>Dykstra splitting for (47)</i>	
Initialize $x \leftarrow y$ , $p \leftarrow 0$ , $q \leftarrow 0$	
While $\neg$ converged, iterate:	
$\begin{cases} e \leftarrow P_1^\psi(x + p) \\ p \leftarrow x + p - e \\ x \leftarrow \Pi_{\mathcal{X}}(e + q) \\ q \leftarrow e + q - x. \end{cases}$	(48)

It can be shown that iterating (48) converges to the solution of (47). In practice, it usually suffices to run Dykstra splitting for a few iterations (2–10) only.